



## Fuzzy Pattern Recognition Based Fault Diagnosis

Rafik Bensaadi, Leïla-Hayet Mouss, Mohamed Djamel Mouss, Mohamed Benbouzid

### ► To cite this version:

Rafik Bensaadi, Leïla-Hayet Mouss, Mohamed Djamel Mouss, Mohamed Benbouzid. Fuzzy Pattern Recognition Based Fault Diagnosis. International Review on Modelling and Simulations (I.RE.MO.S.), 2011, 4 (6), pp.3361-3370. hal-00706004

**HAL Id: hal-00706004**

**<https://hal.science/hal-00706004>**

Submitted on 8 Jun 2012

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Fuzzy Pattern Recognition Based Fault Diagnosis

Rafik Bensaadi<sup>1</sup>, Leïla H. Mouss<sup>1</sup>, Mohamed D. Mouss<sup>1</sup> and Mohamed Benbouzid<sup>2</sup>

---

**Abstract**—In order to avoid catastrophic situations when the dynamics of a physical system (entity in Multi Agent System architecture) are evolving toward an undesirable operating mode, particular and quick safety actions have to be programmed in the control design. Classic control (PID and even state model based methods) becomes powerless for complex plants (nonlinear, MIMO and ill-defined systems). A more efficient diagnosis requires an artificial intelligence approach. We propose in this paper the design of a Fuzzy Pattern Recognition System (FPRS) that solves, in real time, the main following problems: 1) Identification of an actual state; 2) Identification of an eventual evolution towards a failure state; 3) Diagnosis and decision-making. Simulations have been carried for a fictive complex process plant with the objective to evaluate the consistency and the performance of the proposed diagnosis philosophy. The obtained results seem to be encouraging and very promising for application to fault diagnosis of a real and complex plant process. **Copyright © 2011 Praise Worthy Prize S.r.l. - All rights reserved.**

**Keywords:** Diagnosis, fault detection, pattern recognition, fuzzy control, conjugate gradients, complex plant process.

---

## Nomenclature

PID	= Proportional, Integral and Derivative controller;
MIMO	= Multiple Input / Multiple Output systems;
FPRS	= Fuzzy Pattern Recognition System;
FMEA	= Failure Modes and Effects Analysis;
PCA	= Principal Component Analysis;
FCM	= Fuzzy <i>c</i> -Means;
RBNN	= Radial Basis Neural Network;
NP	= crisp Nearest Prototype;
<i>k</i> -NN	= <i>k</i> -Nearest Neighbor;
CUSUM	= CUMulative SUM;
FIR filter	= Finite Impulse Response filter;
pdf	= probability density function;
FIE	= Fuzzy Inference Engine.

## I. Introduction

There is an increasing interest in the development of intelligent fault detection and diagnosis in industrial systems because of increasing requirements for reliable, safe and efficient operation of the plant and for maintaining quality of the products [1-5]. Many variables, unknown or not directly measured, have to be included in the state vector to better describe the plant behavior: model accuracy, a very difficult task, is necessary for the effective processing of unpredictable and imprecise information.

However, human expert can skillfully control plants, localize a fault and in many times make a good diagnosis: the human has the ability to learn, to manage imprecise data and he acts in terms of a complex combination of sensing signals instead of separate information sources.

Because of complexity in modeling a real plant, we need to achieve this sophisticated level of information processing that the brain is capable of, to solve the difficult task of fault detection and diagnosis [6-7].

Pattern Recognition is a field concerned with machine recognition of meaningful regularities in noisy or complex environments. It is based upon the numerical representation of the  $k^{\text{th}}$  object observed in the process (physical entity such as a induction motors, cables, etc.) as a vector  $\mathbf{x}_k = [x_{k1}, \dots, x_{kq}]^T$ , called the *pattern vector* or *feature vector*, where  $x_{kj}$  the  $j^{\text{th}}$  characteristic (feature) associated with observation  $k$ : temperature, pressure, flow, sound noise frequency, etc. and  $q$  the pattern vector length [8-11]. Fuzzy logic concept is included to better manage uncertainty and make useful quantification of hard attributes [12-16].

In this paper, a technique for membership function approximator design is presented. We discuss some classification approaches and apply CUSUM algorithm with additional criterions in fault detection problem [17-18]. We propose a general diagnosis and decision making scheme and give simulation results for a fictive complex system.

## II. FPRS Description

The pattern vector corresponds to a combination of sensing signals: temperature at point A, pressure level at point B, incoming flow, etc. It is constructed in terms of the human expert point of view about the plant, and the effects listed in an FMEA [19]. Other mathematical techniques like PCA help to design the pattern vector [20-21].

For each new incoming observation, we need to identify and quantify the actual plant status and any possible convergence toward another state: in particular, a failure state. We have to estimate the speed evolution and execute the necessary safety actions in acceptable delays. A general fault detection and diagnosis that meet these requirements is presented in Fig. 1.

## II.1 Membership Function Estimation

**II.1.1 Fuzzy clustering.** This first step of unsupervised learning is necessary to produce a logic initialization of the fault detection and diagnosis system. Given the training set  $X = \{x_1, x_2, \dots, x_n\}$ , where  $\mathbf{x}_k = [x_{k1}, \dots, x_{kq}]^T$  the pattern vector, the problem of fuzzy clustering in  $X$  is to assign to the objects  $\{x_k\}$  labels that identify ‘natural subgroups’ in  $X$ . The *membership degrees* are computed as  $U = [u_{ik}]$  by the FCM algorithm with the following considerations [12], [22]:

- A class set of observations that have similar properties, corresponds to one operating or failure mode, the number of clusters  $c$  is assumed to be known. It is also initialized in terms of the expert point of view.
- The training set is considered, as representative of the whole possible clusters, when its size is large enough. It is obtained by causing the plant to operate under different modes.

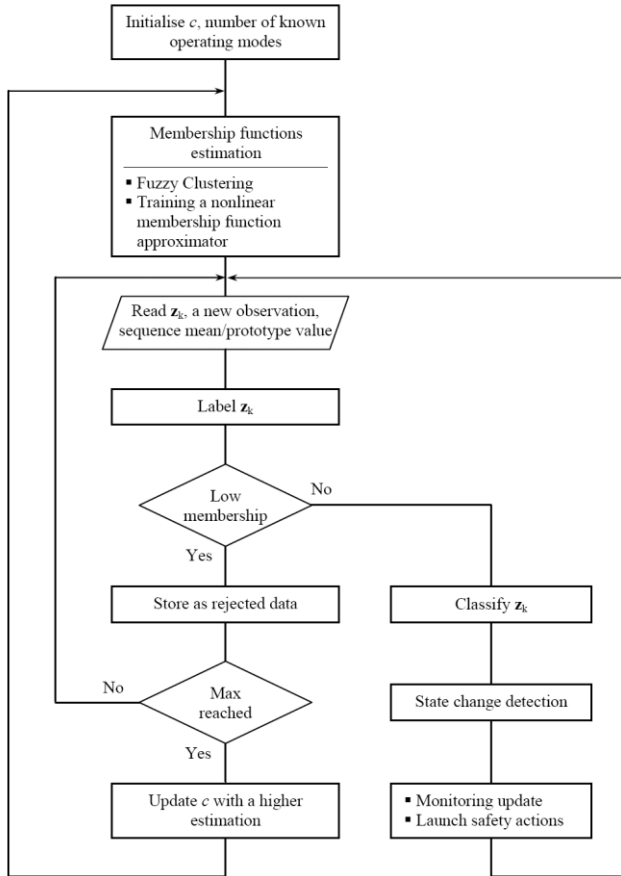


Fig. 1. A general FPRS design strategy.

The FCM algorithm converges from any initialization to a local minimum. The prototypes and membership degrees are iteratively updated by [12]

$$\begin{cases} \mathbf{v}_i = \frac{\sum_{k=1}^q u_{ik}^m \mathbf{x}_k}{\sum_{k=1}^n u_{ik}^m} & \text{for } i = 1, 2, \dots, c \\ u_{ik} = f(x_k, v_i, \{v_j\}, m) \end{cases} \quad (1)$$

where  $u_{ik}$  is the membership degree of object  $\mathbf{x}_k$  to class  $i$ ,  $\mathbf{v}_i$  is the prototype of class  $i$ ,  $m \in [1, \infty]$  is the weight exponent on each fuzzy membership, until an error threshold is reached.

Equation (1) is intuitively understood when we observe the similarity with the *centre of gravity* concept.

**II.1.2 Nonlinear approximator design.** At this step,  $X = \{\mathbf{x}_k\}$  and  $U = [u_{ik}]$  feed the input of a nonlinear approximator optimization algorithm. Let us consider the structure of a RBNN as shown in Fig. 2 [23-24]. The hidden layer typically comprises  $p$  radial basis activation functions with an associated Euclidean input mapping. The output is taken as a linear activation function with an inner product.

The input-output relationship, with  $\mathbf{x} = [x_1, \dots, x_q]^T$ , is given by

$$F(\mathbf{x}, \boldsymbol{\theta}) = \sum_{j=1}^p w_j \exp(-|\mathbf{x} - \mathbf{c}_j|^2 / \gamma_j^2) \quad (2)$$

where  $\boldsymbol{\theta} = [w_1, \dots, w_p]^T$  is the weight vector to be adjusted during learning,  $\mathbf{c}_j = [c_{j1}, \dots, c_{jn}]^T$  are the centers of Gaussian functions.

Now, it is desired to cause  $F_i(\mathbf{x}, \boldsymbol{\theta})$  matching a membership function of class  $i$  at the data points  $(\mathbf{x}_k, \{u_{ik}\})$  for  $i = 1, \dots, c$ , previously estimated by the FCM. Numerous optimization algorithms can be applied for training purposes [25]. In this paper, we focus on the conjugate gradient algorithm for training the approximator. This algorithm can be considered as an intermediate between first- and second-order methods. In [26] a comparative study of five training algorithms reveals that the gradient algorithm have reasonable convergence speeds and require small memory storage. Moreover, the use of Fletcher-Reeves or Polak-Ribiere conjugated gradient methods can avoid main reasons of convergence difficulties such as local minima [27].

The adopted algorithm is based upon the minimization of

$$J_i = \sum_{k=1}^n (e^k)^T e^k$$

where  $e^k = (u_{ik}) - F_i(\mathbf{x}_k, \boldsymbol{\theta})$ , for  $i = 1, \dots, c$ .

The algorithm is given as follow [25], [27].

$$1) \text{ Calculate } \zeta(k) = \frac{\partial J_i}{\partial \boldsymbol{\theta}} \bigg|_{\boldsymbol{\theta}=\boldsymbol{\theta}(k)}.$$

Set the search direction equal to  $d(k) = -\zeta(k)$ .

2) Find  $\boldsymbol{\theta}(k+1)$  which minimizes  $J_i(\boldsymbol{\theta})$  along  $d(k)$ , iteratively, by the Secant method:

- a) Initialize  $\sigma < 1$ , set  $\boldsymbol{\theta} = \boldsymbol{\theta}(k)$
- b) Set  $\alpha = -\sigma \frac{[\zeta(k)]^T d(k)}{[\zeta(k + \sigma d(k))]^T d(k) - [\zeta(k)]^T d(k)}$
- c)  $\boldsymbol{\theta} = \boldsymbol{\theta} + \alpha d(k)$
- d)  $\sigma = \alpha$
- e) If  $|\alpha d(k)| < \text{tol}_\alpha$  then return  $\boldsymbol{\theta}(k+1) = \boldsymbol{\theta}$  else go to b

3) Calculate  $\zeta(k+1)$ .

4) If  $\left| \frac{\zeta(k)}{\zeta(0)} \right| < \text{tol}_\theta$  then return  $\boldsymbol{\theta}(k+1)$

5) Set the next search direction

$d(k+1) = -\zeta(k+1) + \beta(k+1) d(k)$ , where

$$\beta(k+1) = \frac{[\zeta(k+1)]^T \zeta(k+1)}{[\zeta(k)]^T \zeta(k)} \quad (\text{Fletcher-Reeves update})$$

$$[27]) \text{ or } \beta(k+1) = \frac{[\zeta(k+1) - \zeta(k)]^T \zeta(k+1)}{[\zeta(k)]^T \zeta(k)} \quad (\text{Polak-}$$

Ribiere update [27]).

6) Set  $k = k+1$  and goto 2.

$c$  RBNNs are trained to estimate a membership function for each corresponding class. Note that  $F_i(\mathbf{x}, \boldsymbol{\theta})$  may be outside  $[0,1]$  by a very small amount for the first training, because (2) does not include a saturation factor. The few false measures must be corrected (a value that is negative or greater than 1 is taken, respectively, as 0 or 1) to be processed correctly for fault detection. Another procedure, that adds a sigmoid stage to the structure of Fig. 2, can be tried in the future [26].

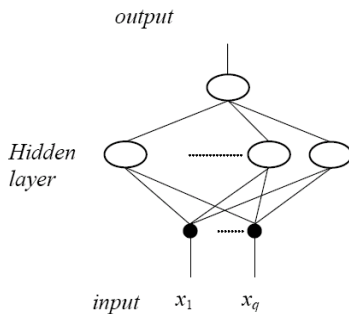


Fig. 2. RBNN based nonlinear approximator.

### III. New Observation Processing

Once the membership approximator is well defined, a new observation  $\mathbf{z}$  is labeled and classified. The membership value of  $\mathbf{z}$  to class  $i$  is

$$\mu_i(\mathbf{z}) = F_i(\mathbf{z}, \boldsymbol{\theta}) \quad (3)$$

We define a hard classifier on  $\mathcal{R}^q$  as a decision function  $\mathbf{D}$  imaged in the canonical (unit vector) basis of Euclidean  $c$ -space so that  $\mathbf{D}(\mathbf{z}) = \mathbf{e}_i$  means that  $\mathbf{z}$  belongs to class  $i$ . This hard attribution is quantified by (3) to explain how much  $\mathbf{z}$  is considered as  $i^{\text{th}}$  fault type and is useful to identify the actual operating/failure mode. There are many choices for classifier design.

Criterion 1:

$$\mathbf{z} \in i \Leftrightarrow \mu_i(\mathbf{z}) = \max \{ \mu_j(\mathbf{z}) \}_{j=1, \dots, c} \quad (4)$$

Criterion 2: Crisp nearest prototype rule (NP rule) [28].

$$\mathbf{z} \in i \Leftrightarrow \mathbf{D}_{\text{NP}, \mathbf{v}}(\mathbf{z}) = \mathbf{e}_i \Leftrightarrow \|\mathbf{z} - \mathbf{v}_i\| \leq \|\mathbf{z} - \mathbf{v}_j\| \quad \text{for } j = 1, \dots, c. \quad (5)$$

Criterion 3: Fuzzy  $k$ -nearest neighbor ( $k$ -NN) rule [29].

– Compute and rank the distances

$$d(\mathbf{z}, \mathbf{x}_i) \text{ as } \{d_1 \leq d_2 \leq \dots d_k \leq d_{k+1} \leq \dots d_n\}.$$

– Find the columns in  $\mathbf{U}$  corresponding to the  $k$  nearest neighbor indices  $\{1, 2, \dots, k\}$ .

– Calculate the vector  $\mathbf{u}^*(\mathbf{z}) = [u(1|\mathbf{z}) \ u(2|\mathbf{z}) \ \dots \ u(c|\mathbf{z})]^T$

$$\text{with the NN labels: } u(i|\mathbf{z}) = \sum_{j=1}^k \frac{u_{ij}}{k} \text{ for } j = 1, \dots, c.$$

– And finally decide

$$\mathbf{z} \in i \Leftrightarrow \mathbf{D}_{\text{NN}, k}(\mathbf{z}) = \mathbf{e}_i \Leftrightarrow u(i|\mathbf{z}) = \max \{ u(j|\mathbf{z}) \}_{j=1, \dots, c} \quad (6)$$

For a long training set and an efficient approximator, the first criterion is the most adequate. NP and  $k$ -NN may be used as a redundant alternative to solve ambiguous situations like the example illustrated in Fig. 3. It is easy to see that  $(\mathbf{z}_1 < \mathbf{z}^* \in \text{class 1})$  and  $(\mathbf{z}_2 > \mathbf{z}^* \in \text{class 1})$ , but we need an additional/other criterion to classify  $(\mathbf{z}_2 \approx \mathbf{z}^*)$ . We add the constraint

$$\sum_{i=1}^c \mu_i(\mathbf{z}) > u_{th} \quad (7)$$

to reject observations with low membership degrees ( $u_{th}$  is a small nonzero number taken lower than 0.5). When a sufficient number of similar (low variance for a Gaussian pdf approximation) observations are reached, a new cluster is created. When a sufficient number of similar (low variance for a Gaussian pdf approximation) observations are reached, a new cluster is created. Prototype and membership function parameters are computed individually (partial FCM with  $c = 1$ ) or by restarting a global membership function estimation process.

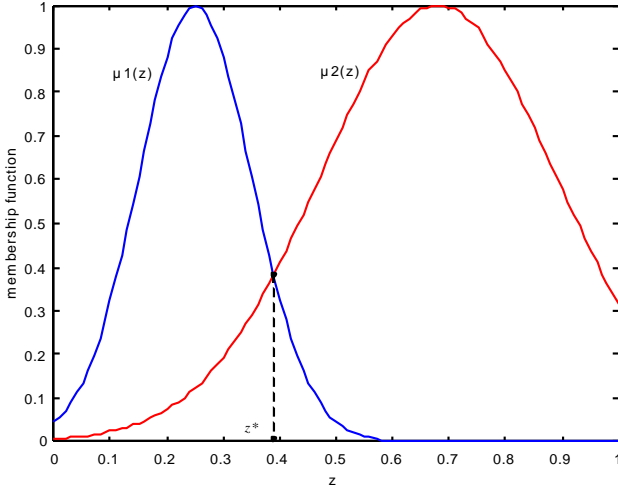


Fig. 3. Example of an ambiguous classification problem.

#### IV. Fault Detection and Forecasts

This is a more ambitious and potentially useful task in maintenance monitoring. The detection of an actual or future operating/failure mode requires getting and processing, in real time, signals  $\mathbf{z}(t)$  and  $\mu_i(\mathbf{z}, t)$ , and taking advantage of their stochastic properties. If the plant status is efficiently described by the pattern vector, we note by  $\mu_i(t)$  the membership degree of the plant state to class  $i$  at time  $t$ , and we develop our approach through the following steps.

**Step 1:** CUSUM algorithm is involved in change detection by processing a sequence of independent random variables with probability density function  $p_{\Theta}(\mathbf{z})$  depending upon one parameter  $\Theta$ . It relies on a fundamental concept: the log-likelihood ratio of an observation  $\mathbf{z}$

$$s(\mathbf{z}) = \ln \frac{p_{\Theta_1}(\mathbf{z})}{p_{\Theta_0}(\mathbf{z})} \quad (8)$$

before an unknown change time  $k_0$ ,  $\Theta$  is equal to  $\Theta_0$ . At time  $k_0$ , it changes to  $\Theta = \Theta_1 \neq \Theta_0$ . The problem is to detect the change time.

The cumulative sum

$$S(k) = \sum_{j=1}^k s(\mathbf{z}(j)) = \sum_{j=1}^k \ln \frac{p_{\Theta_1}(\mathbf{z}(j))}{p_{\Theta_0}(\mathbf{z}(j))} \quad (9)$$

where  $\{\mathbf{z}(j)\}_{j=1, \dots, k}$  a sequence of independent random variables is expected to exhibit a negative drift before change, and a positive drift after change. CUSUM algorithm is derived under this idea and given as follow.

At each sample time:

- Acquire the new data  $\mathbf{z}(k)$ .
- Compute the decision function  $g(k) = \max\{0, g(k-1) + s(\mathbf{z}(k))\}$ .

(c) Compute the number of successive observations for which the decision function remains strictly positive:  $N(k) = N(k-1) \cdot 1_{\{g(k-1) > 0\}} + 1$

where  $1_{\{x\}} = 1$  when  $x$  is true and  $1_{\{x\}} = 0$  otherwise.

(d) If  $g(k) > h$ , issue an alarm ( $h$  is a threshold chosen to meet either a specified mean time for detection or a specified mean time between false alarms). Find the change occurrence time:  $k_0 = k_a - N(k_a)$ , where  $k_a$  is the alarm time. Reinitialize the decision function to 0.

In many practical cases,  $\Theta$  is taken as the mean value of a Gaussian distribution  $p_{\Theta}(\mathbf{z})$ . In our problem, each typical value  $\Theta_i$  indicates a class prototype  $\mathbf{v}_i$ , and the problem of change detection between failure modes will require a prior knowledge about the class-statistical properties. We only own a membership function database.

**Step 2:** Because of the fact stated above, CUSUM will be applied with the following modification:

$$\ln \frac{\mu_i(\mathbf{z})}{\mu_j(\mathbf{z})} \text{ is considered instead of } \ln \frac{p_{\Theta_i}(\mathbf{z})}{p_{\Theta_j}(\mathbf{z})}$$

where  $i$  and  $j$  are class-indexes. A membership value does not have the same meaning as probability, but ratios reflect the same information, so the ability to apply CUSUM with taking

$$s(\mathbf{z}) = \ln \frac{\mu_i(\mathbf{z})}{\mu_j(\mathbf{z})} \quad (10)$$

is intuitively concluded.

**Step 3:** Change time detection between two states is presented. If the target class prototype remains far,  $k_0$  may be considered as an evolution detection occurrence and safety decisions are executed in acceptable delay. When the radius of target class membership function is very small, the safety task will be more difficult, so we need another tool to better quantify the evolution between states and make an earlier alarm. An evolution towards a fault is described by

$$\frac{d\mu_i(t)}{dt}$$

A negative value means that the plant is leaving state  $i$ , a positive value means that it is evolving towards this state. The evolution speed attributes *quick* or *slow* are quantified by

$$\frac{d^2\mu_i(t)}{dt^2}$$

The change in evolution speed is said to be *quick* for values greater than 0, an observation may leave quickly state  $i$  while converging slowly to state  $j$ . Information about the fault evolution direction are extracted from a  $3 \times c$  matrix defined by

$$E = \begin{bmatrix} \mu_1 & \mu_2 & \dots & \mu_c \\ \frac{d\mu_1}{dt} & \frac{d\mu_2}{dt} & \dots & \frac{d\mu_c}{dt} \\ \frac{d^2\mu_1}{dt^2} & \frac{d^2\mu_2}{dt^2} & \dots & \frac{d^2\mu_c}{dt^2} \end{bmatrix} \quad (11)$$

The corresponding alarm time  $k_e$  is computed in terms of constraints on the elements of  $E$ . For example,  $k_e$  may be defined as the delay time for which both

$$\left( \frac{d\mu_i(t)}{dt} \right) \text{ and } \left( \frac{d\mu_i(t)}{dt} \cdot \frac{d^2\mu_i(t)}{dt^2} \right)$$

remain positive, and this corresponds to the alarm time  $k_a$  computed by CUSUM. Other conditions may be added to make an earlier alarm (optimization problem).

Because of external disturbances, a noise is added to  $\mathbf{z}$  when reading. We will consider mean values instead of instantaneous values. The problem is solved by a digital FIR filter, the frequency bandwidth and sampling time are chosen in terms of the noise properties and the response time of all the mechanical/electrical plant parts considered in the diagnosis design [30].

## V. Diagnosis and Decision Making

A fault detection scheme has been completely described. The  $i^{\text{th}}$  fault type effects (symptoms) may be caused by more than one physical entity, and this fact is described by conditional probabilities. Diagnosis is to decide that element  $e_j$  (e.g. an induction motor rotor bar, an inverter switch, etc.) is (or will be) the cause of the detected (or expected) fault. Previous fault events feed a statistical database with class-conditional pdf(s)  $\{p(i^{\text{th}} \text{ fault} | e_j\text{-fault})\}$ , used to compute  $p(e_j\text{-fault} | i^{\text{th}} \text{ fault})$  by Bayes rule [31-32]. The corresponding safety actions are made according to the diagnosis conclusion, the fault severity and the decision making scheme. One powerful solution is built upon an *Inference Engine*: this is a software or hardware system, which gives a *conclusion* (output) from a *fact* (input) and *knowledge* (production rules). If knowledge includes fuzzy linguistic terms, it is referred to as FIE [33]. A conclusion may deal with:

- A new reference tracking (fuzzy control), the knowledge base includes rules of the form:

*if (mode 2) and (low inflow), then (tank 3 temperature should be low)*

- Diagnosis / binary logic instructions, a production rule may be:

*if (water outflow  $> 0.24 \text{ m}^3/\text{sec}$ ) and (valve 21 closed), then (shut-off and repair/change element  $e_2$ )*

*if ( $d^2\mu_3/dt^2 > 0.12$ ) or (input control  $u_1$  not set), then ( $3^{\text{rd}}$  fault type in the next 3 min)*

Beyond the construction/generation of production rules, one difficult task when implementing a fuzzy control algorithm is the accuracy of meaningful membership functions for all the fuzzy linguistic terms considered in the knowledge base [34-35]. We will present later, through an example of temperature control, the different steps involved in fuzzy control implementation.

## VI. Simulation Results

For the demonstration of the proposed diagnosis method, we consider a fictive complex process plant. We assume that a human expert is supervising the plant state by observing three variables:  $v_1$  (pressure at point  $A_1$ ),  $v_2$  (temperature at point  $A_2$ ) and  $v_3$  (sound noise frequency). He makes detection and diagnosis upon two complex combinations:  $x_1 = f_1(v_1, v_2, v_3)$  and  $x_2 = f_2(v_1, v_2, v_3)$  (PCA).

We want to apply the designed FPRS to act with a similar reasoning faculty.

Simulation is run, by causing the plant to operate during a sufficient time, under one normal (typical) operating mode and two failure modes (plant parameters randomly affected). PCA has reduced the pattern vector to  $[x_1, x_2]^T$ . The unsupervised learning step is applied with a training set of 100 data points. Samples are labeled and the prototypes identified as shown in Fig. 4.

The conjugate gradients method is successfully applied to train an RBNN based membership function approximator for each class (Fig. 5).

For classification and fault detection test, we caused the system to evolve towards mode 3 by generating a linear path sequence  $\{\mathbf{z}_k = [z_{k1}, z_{k2}]^T\}$ , each observation is well labeled and classified (Fig. 6a). CUSUM is applied with

$$s(\mathbf{z}) = \ln \frac{\mu_3(\mathbf{z})}{\mu_1(\mathbf{z})} \quad (\text{Fig. 6b}).$$

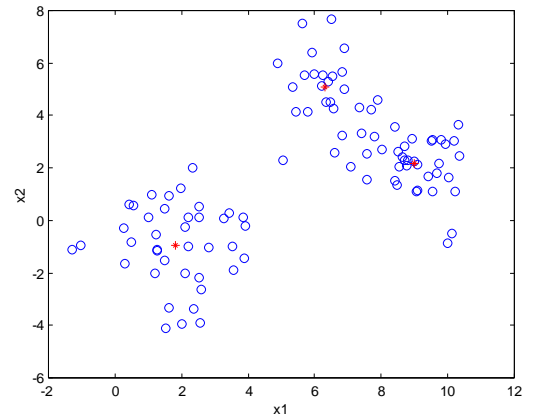
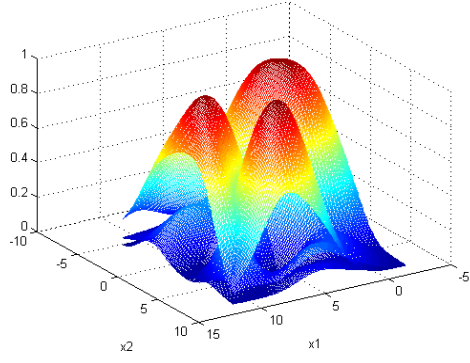
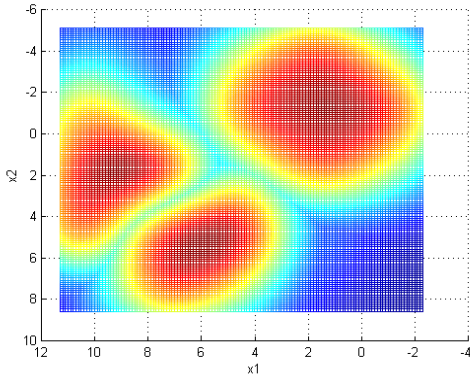


Fig. 4. Fuzzy clustering with  $c = 3$ ,  $q = 2$ . The prototypes are marked as red stars:  $\mathbf{v}_1 = [1.823, -0.935]^T$ ,  $\mathbf{v}_2 = [9.006, 2.151]^T$ ,  $\mathbf{v}_3 = [6.297, 5.078]^T$ .

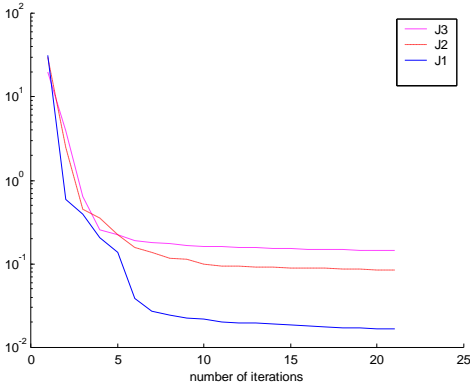




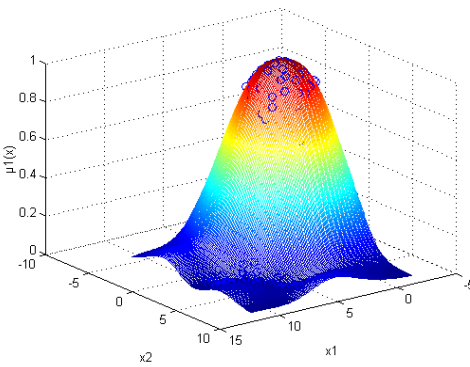
(a) Plant status membership functions.



(b) Projection of (a) on  $x_1$ - $x_2$  plane: Similarity with Fig. 4 is proved.

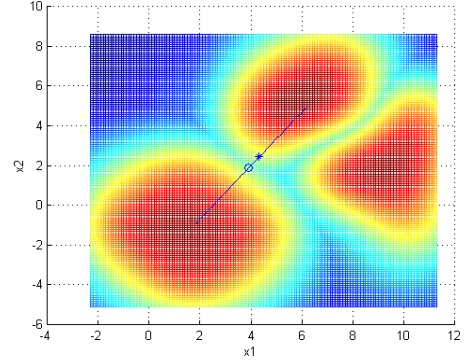


(c) Cost function during learning. There is a trade-off between the learning time and accuracy requirements.

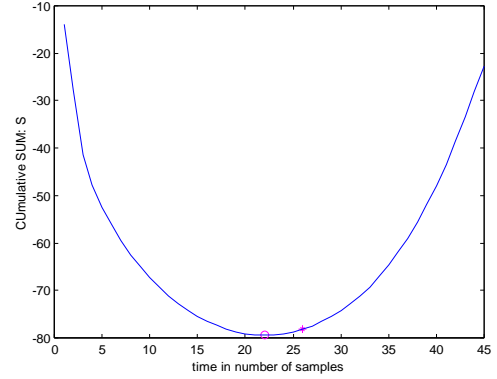


(d)  $F_1(\mathbf{x}, \boldsymbol{\theta})$  matches the data pairs considered in training the RBNN.

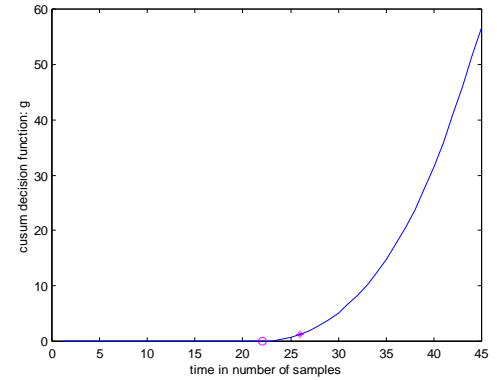
Fig. 5. Membership approximator,  $p = 25$ ,  $\gamma = 2.5$ .



(a) New observation-path, plant is leaving mode1 towards mode 3.



(b) Cumulative sum.



(c) Decision function.

Fig. 6. Fault change detection by CUSUM,  $h = 1.2$ . The estimated change occurrence is marked as circle; the alarm time as star.

Evolution towards fault 3 is detected earlier when membership function derivatives are considered (Fig. 7b).

Temperature control problem is presented to describe an example of a fuzzy inference engine (Fig. 8). A part of the knowledge base is given as follow.

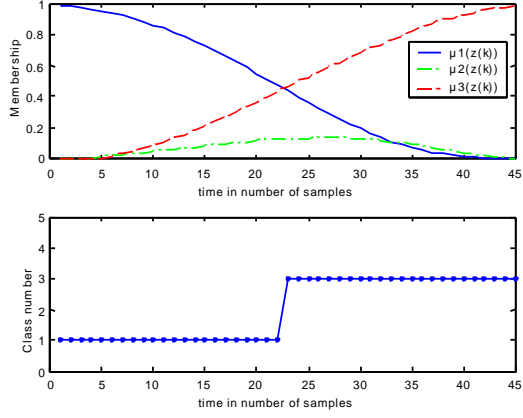
**R1:** *if (mode1) and (quick evolution toward mode3), then ( $T_5$  should be low)*

**R2:** *if ( $P_5 \approx 0.4$  bar) or (slow evolution toward mode3), then ( $T_5$  should be around  $15^\circ\text{C}$ )*

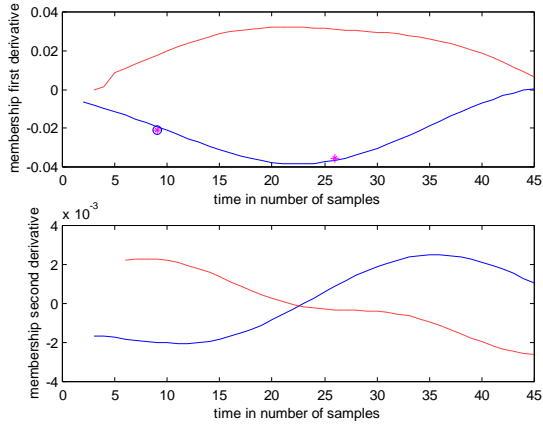
**R3:** *if (mode2) and (high sound noise frequency), then ( $T_5$  should be high)*

**Facts:**  $\mathbf{z} = [7, 3.7]^T$ ,  $P_5 = 1.27$  bar,  $d\mu_3/dt = 0.2$  /sec,  $d^2\mu_3/dt^2 = -0.18$  /sec<sup>2</sup>,  $f_{sn} = 15$  kHz.

**Conclusion:**  $T_5$  should be?



(a) Criterion 1 classification.



(b) 1<sup>st</sup> and 2<sup>nd</sup> derivatives of  $\mu_1(t)$  and  $\mu_3(t)$ .  
The filled circle indicates earlier change detection.

Fig. 7. Future fault detection strategy with additional derivative based criteria.

The fuzzy linguistic term *mode i* is described by the corresponding membership function  $F_i(\mathbf{x}, \boldsymbol{\theta})$ . The membership function for each other fuzzy linguistic term is initialized as shown but may be modified by learning to update the shape form and parameters. The basic operators, involved in fuzzy control, are defined as follow.

$$\text{AND: } \mu_{A \cap B} = \text{MIN}(\mu_A, \mu_B) \quad (12)$$

$$\text{OR: } \mu_{A \cup B} = \text{MAX}(\mu_A, \mu_B) \quad (13)$$

$$\text{NOT: } \mu_{\bar{A}} = 1 - \mu_A \quad (14)$$

For each rule, the compatibility of the fact to the antecedent is obtained by projecting the fact to the corresponding membership function. The resulting membership degrees are combined by a conjunction AND (rules 1, 3) or OR (rule 2). An individual conclusion is obtained by truncating (minimizing) the consequent membership function. All the rules are combined by a conjunction ALSO (maximization of individual conclusions) to construct a relatively complicated membership function  $\mu$  characterizing the final conclusion. The final step is defuzzification.

The new reference  $T_5^*$  that must be tracked, given the facts: ( $\mathbf{z} = [7, 3.7]^T$ ,  $P_5 = 1.27$  bar,  $d\mu_3/dt = 0.2$  /sec,  $d^2\mu_3/dt^2 = -0.18$  /sec<sup>2</sup>,  $f_{sn} = 15$  kHz), is computed by the center of gravity method

$$T_5^* = \frac{\int T_5 \mu(T_5) dT_5}{\int \mu(T_5) dT_5} = 34.8^\circ\text{C} \quad (15)$$

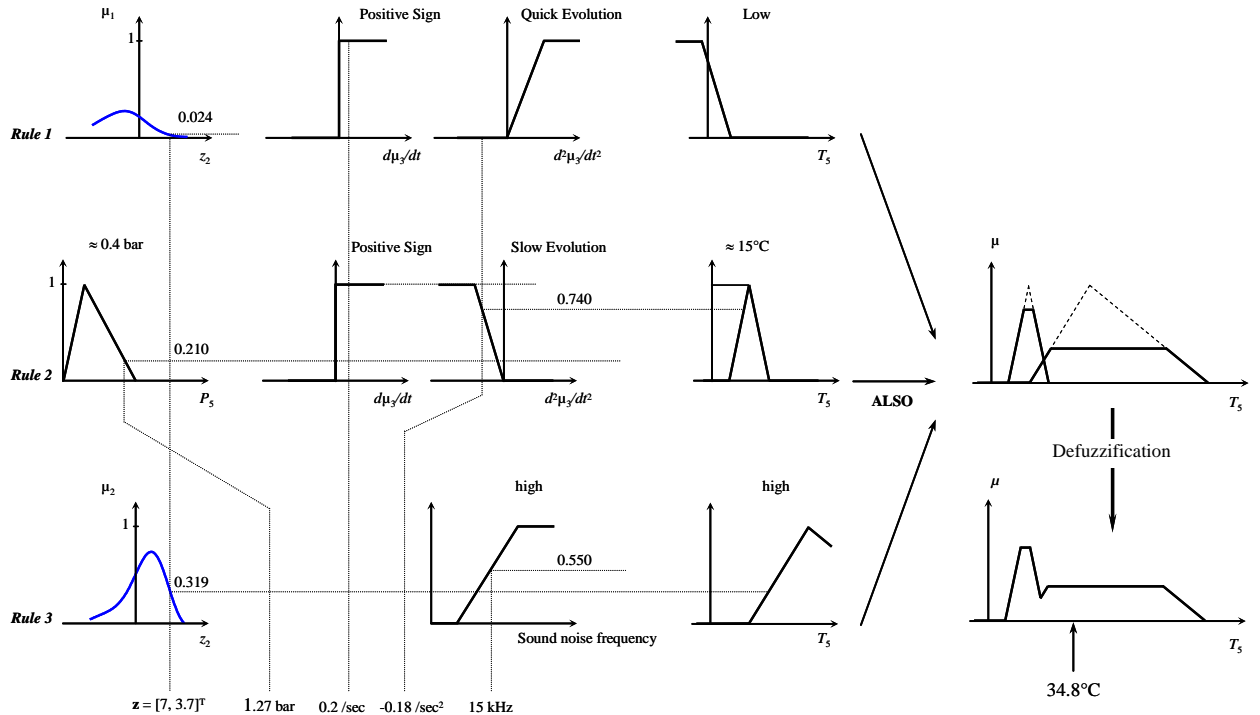


Fig. 8. Example of a fuzzy inference engine based decision process.



and  $T_5$  remains continuously under this control.

## VII. Conclusion

We have proposed a general FPRS design scheme for fault detection and diagnosis in industrial systems. This approach involves fuzzy clustering as a first partition of the training set into a number of classes initialized by the known operating/failure modes, and the conjugate gradient method as the learning tool for training membership function approximator. Incoming observations will be classified and new created classes are taken into account.

Fault detection efficiency is first tested by applying CUSUM with modified expression of the log-likelihood ratio: membership degrees are considered instead of probabilities. Then, another proposed method that takes advantage of membership function derivatives is investigated, evolution towards a fault type target is quantified and safety actions will be executed in acceptable delays.

There are many ways to design the decision system, we proposed a an efficient knowledge based approach and presented a *temperature fuzzy control* as an example of a safety action based on information about fault change forecasts, extracted from the  $E$  matrix.

The designed FPRS was successfully tested for a fictive complex process plant and the obtained results seem to be encouraging for application on a real and complex plant process. The scheduled complex plant is an agro-alimentary production system [4], [36]. However, this will involve additional hardware and software implementation and will be the subject of a future work.

## References

- [1] C.H. Lo, Y.K. Wong and A.B. Rad, "Intelligent system for process supervision and fault diagnosis in dynamic physical systems," *IEEE Trans. Industrial Electronics*, vol. 53, n°2, pp. 581-592, April 2006.
- [2] A. Barigozzi, L. Magni and R. Scattolini, "A probabilistic approach to fault diagnosis of industrial systems," *IEEE Trans. Control Systems Technology*, vol. 12, n°6, pp. 950-955, November 2004.
- [3] S. Gentil, J. Montmain and C. Combastel, "Combining FDI and AI approaches within causal-model-based diagnosis," *IEEE Trans. Systems, Man and Cybernetics, Part B*, vol. 34, n°5, pp. 2207-2221, October 2004.
- [4] L.H. Mouss, D. Mouss, N.K. Mouss and L. Sefouhi, "Test of Page-Hinckley, an approach for fault detection in an agro-alimentary production system," in *Proceedings of the IEEE ACC'04*, vol. 2, pp. 815-818, Melbourne (Australia), July 2004.
- [5] L. Magni, R. Scattolini and C. Rossi, "A fault detection and isolation method for complex industrial systems," *IEEE Trans. Systems, Man and Cybernetics, Part A*, vol. 30, n°6, pp. 860-865, November 2000.
- [6] S. Simani, "Identification and fault diagnosis of a simulated model of an industrial gas turbine," *IEEE Trans. Industrial Informatics*, vol. 1, n°3, pp. 202-216, August 2005.
- [7] S.C. Tan and C.P. Lim, "Application of an adaptive neural network with symbolic rule extraction to fault detection and diagnosis in a power generation plant," *IEEE Trans. Energy Conversion*, vol. 19, n°2, pp. 369-377, June 2004.
- [8] I.Y. Önel and M.E.H. Benbouzid, "Induction motor bearing failures detection and diagnosis using a RBF ANN Park pattern based method," *International Review of Electrical Engineering*, vol. 3, n°1, pp. 159-165, January-February 2008.
- [9] O. Ondel, E. Boutleux and G. Clerc, "A method to detect broken bars in induction machine using pattern recognition techniques," *IEEE Trans. Industry Applications*, vol. 42, n°4, pp. 916- 923, July-August 2006.
- [10] F. Baldini, R. Bartnikas, F.M.F. Mascioli, C. Mazzetti, M. Panella and R. Risica, "Partial discharge pattern recognition by neuro-fuzzy networks in heat-shrinkable joints and terminations of XLPE insulated distribution cables," *IEEE Trans. Power Delivery*, vol. 21, n°3, pp. 1035-1044, July 2006.
- [11] D. Diallo, M.E.H. Benbouzid, D. Hamad and X. Pierre, "Fault detection and diagnosis in an induction machine drive: A pattern recognition approach based on Concordia stator mean current vector," *IEEE Trans. Energy Conversion*, vol. 20, n°3, pp. 512-519, September 2005.
- [12] C. Hwang and F.C.H. Rhee, "Uncertain fuzzy clustering: Interval type-2 fuzzy approach to C-Means," *IEEE Trans. Fuzzy Systems*, vol. 15, n°1, pp. 107-120, February 2007.
- [13] M.S. Ballal, Z.J. Khan, R.L. Sonolikar and H.M. Suryawanshi, "Adaptive neural fuzzy inference system for the detection of inter-turn insulation and bearing wear faults in induction motor," *IEEE Trans. Industrial Electronics*, vol.54, n°1, pp. 250-258, February 2007.
- [14] H.T. Mok and C.W. Chan, "Neurofuzzy networks based fault diagnosis of nonlinear systems," in *Proceedings of the IEEE ICIT'05*, pp. 708-713, Hong Kong, December 2005.
- [15] F. Zidani, M.E.H. Benbouzid and D. Diallo, "Induction motor stator faults diagnosis by a current Concordia pattern-based fuzzy decision system," *IEEE Trans. Energy Conversion*, vol. 18, n°4, pp. 469-475, December 2003.
- [16] R. Mahdaoui, L.H. Mouss, M.D. Mouss and O. Chouhal, "Temporal neuro-fuzzy systems in fault diagnosis and prognosis," *International Review on Modelling & Simulations*, vol. 4, n°1, pp. 436-440, February 2011.
- [17] A. Ilin, H. Valpola and E. Oja, "Nonlinear dynamical factor analysis for state change detection," *IEEE Trans. Neural Networks*, vol. 15, n°3, pp. 559-575, May 2004.
- [18] K.F. Fong and A.P. Loh, "Method for detecting spectral changes in the frequency domain," *IEE Proc. Control Theory and Applications*, vol. 151, n°6, pp. 713-719, November 2004.
- [19] R. Bono, R. Alexander, A. Dorman, K. Yong-Jin and J. Reisdorf, "Analyzing reliability - A simple yet rigorous approach," *IEEE Trans. Industry Applications*, vol. 40, n°4, pp. 950-957, July-August 2004.
- [20] J.F. Martins, A.J. Pires, V.F. Pires, "Unsupervised neural-network-based algorithm for an on-line diagnosis of three-phase induction motor stator fault," *IEEE Trans. Industrial Electronics*, vol. 54, n°1, pp. 259-264, February 2007.
- [21] N. Miskovic and M. Barisic, "Fault detection and localization on underwater vehicle propulsion systems using principal component analysis," in *Proceedings of the IEEE ISIE'05*, vol. 4, pp. 1721-1728, Dubrovnik (Croatia), June 2005.
- [22] L. Yao and F. Chih-Heng, "A hardness measuring method based on hough fuzzy vertex detection algorithm," *IEEE Trans. Industrial Electronics*, vol. 53, n°3, pp. 950-962, June 2006.
- [23] B. Karimi, M.B. Menhaj and I. Saboori, "Robust adaptive control of nonaffine nonlinear systems using radial basis function neural networks," in *Proceedings of the IEEE IECON'06*, pp. 495-500, Paris (France), November 2006.
- [24] M. Mackenzie and A.K. Tieu, "Asymmetric kernel regression," *IEEE Trans. Neural Networks*, vol. 15, n°2, pp. 276-282, March 2004.
- [25] B.M. Wilamowski, "Neural network architectures and learning," in *Proceedings of the IEEE ICIT'03*, vol. 1, pp. TU1-T12, Maribor (Slovenia), December 2003.
- [26] S.L. Phung and A. Bouzerdoum, "A pyramidal neural network for visual pattern recognition," *IEEE Trans. Neural Networks*, vol. 18, n°2, pp. 329-343, March 2007.
- [27] R. Pytlak and T. Tarnawski, "The method of shortest residuals for large scale nonlinear problems," in *Proceedings of the IEEE ACC'03*, vol. 6, pp. 4742-4747, Denver (USA), June 2003.

- [28] J.C. Bezdek, T.R. Reichherzer, G.S. Lim and Y. Attikiouzel, "Multiple-prototype classifier design," *IEEE Trans. Systems, Man and Cybernetics, Part C*, vol. 28, n°1, pp. 67-69, February 1998.
- [29] H. Zhu and O. Basir, "A K-NN associated fuzzy evidential reasoning classifier with adaptive neighbor selection," in *Proceedings of the IEEE ICDM'03*, pp. 709-712, Melbourne (USA), November 2003.
- [30] H.K. Tonshoff, X. Li and C. Lapp, "Application of fast Haar transform and concurrent learning to tool-breakage detection in milling," *IEEE/ASME Trans. Mechatronics*, vol. 8, n°3, pp. 414-417, September 2003.
- [31] M. Bauer, J.W. Cox, M.H. Caveness, J.J. Downs and N.F. Thornhill, "Finding the direction of disturbance propagation in a chemical process using transfer entropy," *IEEE Trans. Control Systems Technology*, vol. 15, n°1, pp. 12-21, January 2007.
- [32] A. Hernandez, D. Hissel and R. Outbib, "Fuel cell fault diagnosis: A stochastic approach," in *Proceedings of the IEEE ISIE '06*, vol. 3, pp. 1984-1989, Montreal (Canada), July 2006.
- [33] C.N. Lu and C.C. Shen, "Estimation of sensitive equipment disruptions due to voltage sags," *IEEE Trans. Power Delivery*, vol. 22, n°2, pp. 1132-1137, April 2007.
- [34] F.H.F. Leung, H.K. Lam, S.H. Ling and P.K.S. Tam, "Optimal and stable fuzzy controllers for nonlinear systems based on an improved genetic algorithm," *IEEE Trans. Industrial Electronics*, vol. 51, n°1, pp. 172-182, February 2004.
- [35] Y.S. Zhou and L.Y. Lai, "Optimal design for fuzzy controllers by genetic algorithms," *IEEE Trans. Industry Applications*, vol. 36, n°1, pp. 93-97, January-February 2000.
- [36] D. Mouss, L.H. Mouss and N. Mouss, "DIAPASE: an industrial diagnosis tool by knowledge based approach for an agro-alimentary production system," in *Proceedings of the IEEE ACC'04*, vol. 2, pp. 863-868, Melbourne (Australia), July 2004.

<sup>1</sup>University of Batna, LAP, 1, Rue Chahid Boukhlof Mohamed El Hadi, 05000 Batna, Algeria (email: [hayet\\_mouss@yahoo.fr](mailto:hayet_mouss@yahoo.fr)).

<sup>2</sup>University of Brest, EA 4325 LBMS, Rue de Kergoat, CS 93837, 29238 Brest Cedex 03, France (e-mail: [Mohamed.Benbouzid@univ-brest.fr](mailto:Mohamed.Benbouzid@univ-brest.fr)).



**Rafik Bensaadi** received the B.Sc. degree in Electronics and Control Systems and the M.Sc. degree in Industrial Engineering, from the University of Batna, Batna, Algeria, in 1999 and 2006, respectively. He is currently working toward the Ph.D. degree at the University of Batna, Batna, Algeria.

His research interests include machine learning, control theory, microcontroller-based digital control design, DSP design, and hardware implementation of fault detection, diagnosis and decision-making scheme.



**Leila Hayet Mouss** was born in Batna, Algeria, in 1954. She received the B.Sc. degree in Electrical Engineering, in 1979, from the National Polytechnic School of Algiers, Algeria; the M.Sc. and the Ph.D. degrees in Electrical and Computer Engineering from the University of Bordeaux, Bordeaux, France, in 1982 and 1985, respectively, and the University Habilitation degree from the University of Batna, Batna, Algeria, in 2005.

After graduation, she joined the University of Batna, Batna, Algeria, where she is an Associate Professor of Electrical and Computer Engineering. Dr. Mouss is the head of the Automatic and Computer Integrated Manufacturing Laboratory (LAP). Her current research interests include diagnosis of industrial production system using artificial intelligence techniques.



**Mohamed Djamel Mouss** was born in Batna, Algeria, in 1956. He received the B.Sc. degree in Electrical Engineering, in 1982, from the National Polytechnic School of Algiers, Algiers, Algeria; the M.Sc. degree in Industrial Electricity and the Ph.D. degree in Industrial Engineering, both from the University of Batna, Batna, Algeria, in 1989 and 2006, respectively; and the University Habilitation degree from the University of Batna, Batna, Algeria, in 2008.

After graduation, he joined the University of Batna, Batna, Algeria, where he is an Associate Professor of Electrical and Computer Engineering. Dr. Mouss is the leader of the S<sup>3</sup> team (Sûreté, Supervision, Surveillance) of the Automatic and Computer Integrated Manufacturing Laboratory (LAP). His current research interests include diagnosis of industrial production system using artificial intelligence techniques.



**Mohamed El Hachemi Benbouzid** was born in Batna, Algeria, in 1968. He received the B.Sc. degree in electrical engineering from the University of Batna, Batna, Algeria, in 1990, the M.Sc. and Ph.D. degrees in electrical and computer engineering from the National Polytechnic Institute of Grenoble, Grenoble, France, in 1991 and 1994, respectively, and the Habilitation à Diriger des Recherches degree from the University of Picardie "Jules Verne," Amiens, France, in 2000.

After receiving the Ph.D. degree, he joined the Professional Institute of Amiens, University of Picardie "Jules Verne," where he was an Associate Professor of electrical and computer engineering. Since September 2004, he has been with the Institut Universitaire de Technologie of Brest, University of Brest, Brest, France, where he is a Professor of electrical engineering. His main research interests and experience include analysis, design, and control of electric machines, variable-speed drives for traction, propulsion, and renewable energy applications, and fault diagnosis of electric machines.

Prof. Benbouzid is a Senior Member of the IEEE Power Engineering, Industrial Electronics, Industry Applications, Power Electronics, and Vehicular Technology Societies. He is an Associate Editor of the IEEE TRANSACTIONS ON ENERGY CONVERSION, the IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, and the IEEE/ASME TRANSACTIONS ON MECHATRONICS.